SYSTEMS BUILDING; A PERT TIME APPLICATION

by

Leonard Philip Guy

SYSTEMS BUILDING: A PERT TIME APPLICATION

BY

LEONARD PHILIP GUY, III
BS, United States Naval Academy
(1967)

Submitted in partial fulfillment
of the requirements for the degree of
Master of Science
at the
Massachusetts Institute of Technology
(September, 1969)

ABSTRACT

SYSTEMS BUILDING: A PERT TIME APPLICATION

by

LEONARD PHILIP GUY, III

Submitted to the Department of Civil Engineering on August 18, 1969, in partial fulfillment of the requirements for the degree of Master of Science.

The research presented in this thesis is a study of the feasibility of transferring certain aerospace systems technology to the field of urban systems. The fiscal outlays in the aerospace and defense industries has made possible significant advances in technology. Even though the results of these technological advances have been rewarding in their intended applications, it is believed that their potential in other areas has not been exploited fully.

This thesis presents a typical example of aerospace systems technology applied to a construction problem. Specifically, a computer program, NASA PERT TIME II, is used to schedule the construction of a multistory office building. The flexibility and usefulness of this computer program is shown by example.

One conclusion of this study is that with minor effort certain recent technological advances in the aerospace and defense industries can be usefully applied to areas not related to aerospace or defense projects. In particular PERT TIME II, an advanced network analysis tool developed by NASA, holds promise for direct application to building construction projects.

Thesis Supervisor: Russel C. Jones

Title: Associate Professor of Civil Engineering

ACKNOWLEDGEMENTS

The author wishes to express his gratitude to the following persons and organizations for their contributions to this thesis.

Professor Russel C. Jones, thesis supervisor, whose interest was keen, and whose guidance was always available.

Mr. Eugene Pallat, and Mr. Fred Hills, of NASA, for help in obtaining and debugging the computer program.

Mr. James N. Jackson, research assistant, for help in setting up the computer program.

Mr. Calvin F. Opitz, research associate, for help in formulating the network for the New England Merchants Bank Building.

Gay, my wife, typist, and chief worrier, who helped keep my spirits high and my pen writing.

The M. I. T. Urban System Laboratory for partial funding of this project.

This work was done in part at the Information Processing Center, M. I. T., Cambridge, Massachusetts.

And finally, the United States Navy for giving me the opportunity to attend graduate school.

TABLE OF CONTENTS

CHAPTER I

INTRODUCTION

In the short time that the Department of Defense has been in existence, it has made numerous management developments. Probably the most significant development is systems management.[1] Systems management, systems analysis, systems development, or any of the myriad of similar terms can all be grouped into the broad title systems engineering methods. A system is an integrated collection of jobs or activities which lead to a system or project goal. The systems engineering method recognizes this interrelationship of activities, and recognizes the need to optimize the overall system functions. The system functions are usually time, cost, performance, and/or reliability. Further the overall optimum might not coincide with any of the activity optimums.

The corollary to the systems engineering definition is the method of problem solution. A complex system is reduced to a series of activities, the interrelationships located or defined, then the problem functions are optimized. The methods of solution are many. Some will be presented as background and one will be presented in detail in this study.

The key word in the systems approach is performance. In the final analysis it is how one system performs against another or against no system that counts. Performance requires not only identification and definition of system goals, but also the realization of those goals within cost and/or time limits. Performance is therefore a measurement and an evaluation of objectives fulfillment.

The identification of objectives is sufficient justification for the use of the systems approach. Identification of objectives requires estimates of systems functions. This phase may then lead to the realization that in order to meet a deadline more resources must be utilized, or that the system objective can be realized sooner if some flexibility is sacrificed. Such an analysis may lead to the realization that the system objectives can be achieved in less time and/or at a lower cost. All of the above examples involve tradeoffs. Exploring tradeoffs, the various avenues of approach, involves evaluating the systems performance. This step is crucial to decision making. Choosing the best approach within the functional constraints is optimization of the objective.

The recent developments in systems engineering have come primarily from the defense and aerospace organizations. This is due to two factors: the large scale availability of resources, and the size of the tasks undertaken.

These systems developments have not only advanced the needed technology in their intended military and aerospace areas, but have had significant impact in the growth and development of other industries. Two areas of aerospace systems technology show promise for rapid adaptation to problems in the civilian sector:

1. Methodology for handling of complex problems, and
2. developed hardware and computer software.

Since urban construction problems are extremely large and complex, this systems methodology should assist in arriving at better solutions. In some cases already available computer software may also be useful. The adaptation should accelerate progress in the technologically limited area of urban systems. The limitation arises because the funds available for urban systems development have not been sufficient to meet the ever rising number of problems.

Two distinct concepts of the applications of systems methodology are developing within the construction industry: systems approach, and systems building. The first, systems approach, is developing along the classical lines of systems engineering methods as already presented. The second, systems building, on the other hand, works in reverse to the classical methods. Performance specifications are determined for the individual components or activities. The completion of these components, to specifications, yields the predetermined performance for the entire system.

A systematic evaluation of the translation of appropriate technology is in order. In this way maximum usage of recent developments can be obtained. Perhaps the most rapid means of transfer can be obtained by improving existing technology. The developments in network analysis fall into this category.

A network is an ordered schedule of the activities that make up a system. As an example, suppose the system is the erection of a building. A crude network might include, in order: purchase a site, design the building, contract for labor and procure materials, and erect the building. These descriptions are the activities of the network. More will be said about activities later.

Network analysis is a fairly recent technique, having its origin in 1956-1957. Originally called Program Evaluation and Review Technique (PERT), it was developed by the management consulting firm of Booz, Allen, and Hamilton for the United States Navy's Special Project Office. The need for PERT arose out of the complex production requirements of the _Polaris_ nuclear-powered submarine project. The goal of the PERT development was to keep the _Polaris_ project on schedule and within the cost budget.[10]

PERT originally had three time estimates associated with each individual activity. An optimistic time, a most likely time, and a pessimistic time. The assumptions for each are: the probability of finishing earlier than the optimistic time or later than the pessimistic time is one per cent each, and the probability of finishing by the most likely time is fifty per cent.

Network analysis now has many variations and names. The Critical Path Method (CPM) differs from PERT in that there is only one time estimate for each activity, the most likely time. PERT/COST and PERTCO provide for a linear distribution of the activities cost over that activity. PERT TIME has one time estimate for each activity as well as a linear distribution of each activities cost over the period that that activity is being pursued. The difference between PERT TIME and PERT COST is due to slack, which is time in which work could be done but is not. PERT COST distributes cost over the whole activity time whether or not work is physically being done.

The construction industry has been using some form of network analysis for several years. However, the industry is conservative and change takes time. Some of the more recent advances have not yet been added to the network analysts' arsenal of probelm solvers. As originally developed PERT was primarily a scheduling tool. And it has been found that jobs employing PERT do have better cost and schedule performance than jobs that do not use PERT.[7] A fringe benefit lies in communication, in that communication and cooperation is improved between groups involved in a PERT controlled project.[8] This improved communication is the result of well defined objectives and an orderly way of accomplishment.

A project may now be divided into many systems, with each system divided into subsystems and each of the subsystems made up of activities. In this way a whole hierarchy of objectives may be established. Each subsystem may now have an objective. The subsystems are integrated at interfaces. Interfaces are identifiable accomplishments common to two or more subsystems.

With the widespread availability of computers, many of the recent advances have been in computer software. Also, with larger computers, much larger systems can be analyzed by computer methods.

The National Aeronautics and Space Administration has developed an advanced PERT TIME program which they call NASA PERT TIME II. It is a second generation computer program written in FORTRAN IV. This program can easily be adapted in the building industry. A time-sharing version is currently being written to provide for even further usage.

The purpose of this thesis is to show the ease of adaptation and present information necessary to use the

NASA program in a typical construction operation. Development will proceed from the basic PERT network theory to capabilities and program analysis using PERT TIME. A user's manual will be presented in an appendix.

This introduction has been a development of systems engineering methods, proceeding from an overview of the systems approach through the method of PERT TIME. A detailed presentation of PERT TIME will follow.

CHAPTER II

PERT TIME

PERT network analysis is basically a scheduling tool. One of the results of a PERT analysis is the networks critical path. Activities on this path are pacing activities, that is if a pacing activity takes longer to complete than anticipated, the final objective will be delayed. By identifying the pacing activities, potential bottlenecks are foreseen. In this way schedules can be changed to avert, if possible, potential delays. The construction of a network and calculation of its parameters will be shown below. Before developing an example network, a discussion of PERT usage and problem solution will be given.

PERT is widely applicable in several areas of systems engineering. In the following areas PERT has found significant usage.

1. Scheduling projects and programs.
2. Evaluating existing schedules as work progresses.
3. Estimating cost for proposed projects.
4. Evaluating cost versus budget as work progresses.
5. Planning and smoothing resource utilization.

Some of the fields in which PERT has been successfully employed are (1) design and manufacture of weapons systems, (2) implementation of automated information systems, and

(3) preventive maintenance. The simple stepwise procedure used in PERT network analysis is one reason why PERT has been so widely used.

The systems approach generally follows the following steps:

1. Define the problem and determine end objectives.
2. Establish a plan and specify the system requirements.
3. Program the utilization of resources.
4. Execute the activities according to plan.
5. Report, and control the execution by evaluating the feedback.
6. Update the plan as necessary for corrective action.
7. Review the project for information useful on future projects.

Following these steps an example network will be generated.

II.1 PERT Elements

The elements in a PERT network are few in number and their definitions are straight forward.

1. An event. Events are identifiable points in time. An event requires no expenditure of resources and does not take time to take place. Events indicate that something is about to happen and/or that something has happened.

2. An _activity_. Activities are definable tasks to be completed. Activities usually require expenditures of both resources and time. An activity takes place between two events. These events denote the start and the end of the activity, and are referred to as the predecessor event and the successor event, respectively. An activity is referred to by its predecessor and successor event numbers.

3. _Time estimates_. As mentioned previously, there are three time estimates associated with an activity. Since the program to be presented is a PERT TIME program, only the most likely time estimate is used.

4. _Expected time_. Expected time is a forward time calculation. That is, starting at the first event add the most likely times of all activities on a path leading to an event. (A path is a sequential or connected set of activities.) Do this calculation for all possible paths from the start to that event. The path with the longest time duration is the expected time for that event. An expected time for the start of an activity is simply the expected time of that activity's predecessor event. This is usually the first network calculation made.

5. _Allowable time_. This is a backward time calculation. Allowable time is calculated in the same way as expected time except that it starts at the end and activity durations are subtracted. Again the path with the longest

time duration from the end leading to an event determines the event's allowable time. The allowable time for the end event can either be the event's expected time or an assigned schedule date. An activity's latest allowable start is calculated by taking the activity's successor event's allowable time and subtracting the activity's most likely time.

6. Slack time. An activity's slack time is defined as the difference of the activity's expected time and its latest allowable time.

7. Critical path. The critical path or pacing path is the longest path in the network. Activities on the critical path can be identified as having the smallest slack time in the network. Events on the critical path have the same expected and allowed time unless the scheduled date of completion does not coincide with the expected date of completion.

This list, although not complete, is now sufficient for our present purposes. There are other calculations that can be made, but this list contains all of the elements usually used. Using these elements a simple network will be generated.

The first step is the definition of an end objective in clear and precise terms. As shown in Figure 1, the end objective must be exactly defined in order to identify an event to correspond to the end objective.

Next one must define all activities immediately precedent to the end objective. The circles in Figure 1 represent events to be numbered later.

Taking the activities A, B, and C, one at a time, one must then define all activities immediately precedent. It may be discovered that an activity is precedent to two or more activities, then a <u>dummy</u> activity must be placed in the network. In Figure 2, activity D is precedent to activities A and B. Activity D may be placed before either A or B and a dummy activity from activity D's successor event to the other activity's predecessor event is placed in the network. This dummy activity requires no resources or time, it only shows dependence or restraint.

In this way one works backwards until the start is reached. At this time the events can be numbered. Some PERT systems require that event numbers be in increasing order from start to finish along any path. Most likely time estimates may now be made for each activity. This estimate should be as exact as possible, since all calculations are based on activity most likely times. The expected times may now be calculated. Proceeding backwards the allowable times and the slacks are calculated. A scheduled time for any event may be made at any time. Whenever an event is given a scheduled time new calculations must be made for allowable time and slack. Finally the critical path is located.

The above procedure is followed on all PERT networks. A description of a recent computer version of PERT will now be presented.

II.2 Recent Developments

The National Aeronautics and Space Administration (NASA) has over the past several years developed a sophisticated PERT TIME computer program. Taking a program written in machine language by Lockheed Aircraft Corporation in 1963, NASA wrote a version of PERT TIME in FORTRAN IV. FORTRAN IV was chosen because it is currently the most widely used compiler language, and FORTRAN IV is easy to modify. The current program is known as NASA PERT TIME II, hereafter referred to as PERT TIME II. PERT TIME II is a program which can monitor time, cost, and performance, in addition to providing schedules arranged by slack, predecessor event, successor event, organization (if more than one organization is working on a network), expected date, and allowable date. PERT TIME II has a capability of monitoring a network of over 100,000 activities. This capacity is realized by using a modular or subnet technique. Figure 3 shows a subnetted network. The solid line encloses one subnet and the dashed line encloses another subnet. The events with an X in them are common to both subnets. These events are called interface events. The subnet technique allows a division into logical units, such as work done by different tradesmen, or work done on different floors of a multistory building.

There are many advantages of PERT TIME II over earlier PERT programs. The subnet idea is perhaps the most significant advantage. PERT TIME II allows networks to be divided up into subnets. The subnets are coupled at interface events. Interface events are events which are common in two or more subnets. Basic PERT allowed only one network, thus limiting the number of activities in order to keep the calculation time within reason. PERT TIME II has a capacity of 50 subnets of 2,000 activities each.

Another advantage of PERT TIME II is the summary subnet. This subnet is generated by declaring certain events as interfaces. These events are called milestone events, and although they are interfaces they are not necessarily common to more than one subnet. The summary network is made up of all the interfaces, both regular and milestone event interfaces. This subnet is useful to the project supervisor, who does not have to know when every activity is complete, only when certain significant events are reached. The computer program generates the activity durations and resource expenditures for the summary subnet.

Another advantage of PERT TIME II is the method of file maintenance. In previous PERT programs the master file was just a tape bearing the activity cards. A change in the network required that first the master be updated and then a run was made for the whole network. PERT TIME II updates the master as part of the normal PERT run. This eliminates duplication of effort. The master file contains all information generated by a PERT run. This ends the need of recalculating over an entire network. The information is stored by subnets, since for a given run the total network may not be needed. The master file is read only once and the updating and recalculation of a subnet are overlapped.

A final advantage in PERT TIME II is deletion of completed activities from the master. This is accomplished by a control card option and has a twofold effect. First, by elimination of completed activities which no longer affect the project schedule, the effective in-core size of the network is reduced. Second, a large amount of updating is required for removing completed activities, and this type of routine updating is now completely eliminated.

PERT TIME II is suited for solving the systems problems involved for a building system. While PERT is being used in the construction industry, it is believed that the PERT TIME II computer program can be a significant improvement to the existing state of the art.

The development of PERT from its beginning in 1956 through PERT TIME and other PERT forms has been described above. A specific computer program, PERT TIME II, has been discussed in some detail. The program's capabilities and an example problem will now be presented.

CHAPTER III

PROGRAM CAPABILITIES

The capabilities of the PERT TIME II program will be described in this section, including the program's internal logic. The internal logic will be considered through description of individual subroutines responsible for certain tasks.

In order for new technology to be accepted, it must be demonstrated as an improvement over existing methods. The "real world" application was foremost in the development of this program. The major features of PERT TIME II are summarized below.

The program has been written in FORTRAN IV compiler language so that it could be used on the most popular types of hardware configurations. The program may be stored on tape or disk, so that only the control cards for an individual run need to be input. Currently a time sharing version is being developed to make the program available to smaller organizations.

The subnet or fragnet concept allows great flexibility in the system. The division of the network may be by contract, organization, component system, work crew, union, and so forth. Output reports can be generated in parallel with the work breakdown structure. There are fourteen standard output options.

The calendar routine is based on a five day workweek, and two holidays each week. There is a single time estimate for each activity. Whenever a PERT run is made a report date, which is usually the current calendar date, is input to the program. There are four output cut-off options available. Output cutoff allows the user to specify a certain future date or number of weeks past the reporting date after which reports will not be generated. This option allows the user to receive reports for a certain time span, such as, the three months following the report date.

BASIC PERT allows only one starting point and one finishing point. The PERT TIME II program can have as many as 500 starting events and an unlimited number of finishing events. A start event is any event that does not require any precedent activities to be completed. An end event is any event whose completion is not required before an activity may start. The program internally turns all start and end events into interfaces. These interfaces are only common to one subnet. There can only be a total of 500 start, end, and regular interfaces in any subnet.

The program internally generates a master file. This master file contains all the calculated network data. On subsequent reporting runs, the reports can be generated without recalculating all the data. The master file thus greatly reduces computer time on a project requiring multiple runs.

As mentioned before a summarization option cuts down on the output required by a project supervisor. Individual subnets may be extracted and processed alone, disregarding the effects of the other subnets.

The program processes the input one card at a time. The data is read in and stored. After all cards have been read the required output is generated. The program utilizes 48 subroutines to perform specific tasks. The functions of some of the subroutines will be presented next, along with program analysis.

III.1 Program Analysis

The program can be divided into four different parts by considering control phases. These are network analysis, execution control, reporting, and updating. Network analysis begins by condensing all subnets to obtain a control network. The control network consists of all interfaces, starting events, and ending events. All paths are condensed between interfaces to yield one control activity. The control activity is the sum of all the activities' most likely times on the longest duration path between two interfaces. Then the program calculates expected and allowed times for the control network, and determines the expected and allowed dates for each subnet report asked for.

To start, all the interface and activity cards are read for a particular subnet. Interface cards define events which are common to two or more subnets. The activities are defined by a predecessor and a successor event, and each activity has an associated time estimate. Each activity may have a description, date (schedule or actual), and other information such as cost or manhours.

Subroutine TEST locates all start and end events. Start and end events are not to be confused with predecessor and successor events. The former mark the events which have only activities originating from them or events which have only activities ending at them and the latter mark the beginning and ending of individual activities. Subroutine TEST converts all start and end events into interfaces if they are not so already. Subroutine TOPOL is called to map out all paths, and to make time calculations as it goes. Subroutine TSUPE calculates expected times along all paths to each event. The expected time replaces an earlier calculation only when it is greater than before. The reason for this is that only the most pessimistic expected time is desired. Subroutine TSUPL calculates allowable times along all paths to each event. The allowable time replaces the earlier calculation only when it is smaller than before. The reason for this is that only the most optimistic allowed time is desired.

It is of interest to note at this time, expected and allowable times for subnet activities cannot be determined since they are affected by other subnets at the interfaces. However, expected and allowable times are calculated on activities between two interface activities. These times are those associated with the control network activity between the two interface events. Thus, at this time we have two lists: a list of relative times for each subnet event, and a list of control network activities. The relative times are filed with each subnet. As this process proceeds from subnet to subnet, the control network is defined.

The control network is a complete subnet just as any other. However, it is generated internally using all the interfaces as events. The analysis of this subnet proceeds in the same way as the others. The expected and allowable times are those for the interface events.

Now, on subnets specified, reports may be compiled. Expected and allowable times for any event in any subnet can be determined by combining the earlier results of the same subnet and the results of the control subnet.

III.2 Execution Control

Control will be more fully described in the user's manual but the general flow of logic will be instructive here.

The main routine is called ASKER and gets its instructions by reading asterisk control cards. A typical program would be executed as shown in Figure 4. The asterisked cards are read one at a time and subroutine ASKER then executes the appropriate steps before reading another asterisked card. The *TITLE and *DATE cards define the title and date of the network and are stored accordingly. The *SUBNET card informs subroutine ASKER that a subnet follows and the name on the *SUBNET card is stored in a list with all the subnet names in the order read. The interface cards are read first and stored accordingly. The *NETWORK card marks the end of the interface cards and the beginning of the activity cards for the given subnet. Subroutine TEST analyzes the subnet and then returns control to subroutine ASKER. An END card marks the end of the activity cards. This procedure continues until all subnets have been analyzed.

The end of subnets is determined when a card which is not a *SUBNET card is read. In this case it is a *REPORT card. When *REPORT card is read subroutine ASKER calls three sorting subroutines, PREPAR, SSORT, and MOVE. These subroutines analyze the control network. After this is done, control returns to subroutine ASKER, which interprets what was on the *REPORT card previously read. The type of report specified on the given subnet (s) is stored. This process continues until a card which is not a *REPORT card is encountered. Subroutines TEST and TOPOL then furnish the reports asked for. Control then returns to subroutine ASKER, which interprets the card previously read, in this case an *END BATCH card. The execution is terminated.

III.3 Reporting

The specific reports are described in detail in the user's manual in Appendix C. Reports can be generated by sorting predecessor event numbers, sorting successor event numbers, sorting by slack, sorting by expected date, sorting by allowed date, and sorting by organization. Subroutine SUPER determines how the sort is to be made and is called upon completion of the subnet analysis.

Sort subroutines PRSCAN and SORT are called by subroutine SUPER to determine the reordering of the table stored by predecessor. A note should be made that the table is never rearranged, it is duplicated in an activity buffer and subroutines PRSCAN and SORT output from buffer according to control issued by subroutine SUPER. Upon completion of a report, the buffer can be reloaded with a new subnet or the same subnet to be outputted still another way. Subroutine OUTPUT outputs the buffer and formats it. As subroutine OUTPUT requires the date, subroutine FIND locates the required date.

III.4 Updating

The master file developed by the program contains, for each subnet, the subnet name, a flag to distinguish between subnets and summary subnets, all interface cards, tables of relative times calculated for every event in the subnet, control activities and time durations, and activity cards.

Updating consists of any modification to interface cards, activity cards, and subnet summary declarations. In addition, entire subnets may be added or deleted.

The procedure starts when update cards are encountered in an input deck. Subroutine ASKER searches the master file for the update subnet, the information is copied onto the new master file. Update cards are required to be in sequential order in the input deck. Therefore, if the subnet is not encountered on the update subnet, the information is still accurate. This copying is continued until the update subnet is encountered, then the control transfers to subroutine UPDATE.

Subroutine UPDATE reads all interface update cards and sorts them by calling subroutine USORT. This is compared to the master file, and the master file is then updated accordingly when matching interface cards appear.

Subroutine ACTMOD updates the activities on the master file. Activities are stored by predecessor event number on the master file. The first card column on the activity update card is read to determine the code. A "1" in column 1 establishes a new activity. If there is a card on the master file with the same predecessor and successor event numbers, it is deleted and the new card added. A "2" in column 1 indicates a change of an existing activity card's data, i.e., time or resources. A "3" in column 1 indicates a completed activity. The date on the completed activity card is noted in the master file. A "4" in column 1 indicates that the date is a scheduled start date for that activity. A "5" in column 1 deletes the activity from the master file.

Subroutine ADD contains the new subnet as it is being updated. Upon completion of a subnet subroutine ADD will take the original, the update, or an altered subnet and call subroutine TEST to perform a new analysis. After analysis subroutine TEST places the new information onto the new master file. Then, as usual, the control subnet is analyzed and reports output as required.

The flexibility of PERT TIME II has been explored in this section. It is clear that the internal logic makes this program highly adaptable to projects outside of the aerospace industry. This logic proceeds stepwise, that is, as an individual task is recognized directing subroutines, such as TASK and TRANS, call the appropriate analysis subroutines. In this way a wide variety of tasks can be undertaken. Report generation has been presented in summary form to indicate the types of reports available. The updating technique has been presented to show the time saving features on calculations made for multiple runs. An example problem utilizing these capabilities of PERT TIME II will be described in the following chapter.

CHAPTER IV

EXAMPLE: NEW ENGLAND MERCHANTS BANK

In this section an example construction project is described. The network is generated for the New England Merchants Bank Building in Boston, Massachusetts. The building, which has already been constructed, has been chosen because of the availability of data. The example runs will show the ease of use of PERT TIME II, the various forms of output, and the applicability in the construction industry.

The network generated consisted of a main subnet, which contained the vertical work, such as, structural steel, ventilation ducts, plumbing, elevators, and granite, for the whole building and fourteen tier subnets. A tier is three floors, except for the fourteenth tier, which is one floor. The tiers contain the horizontal work, such as, wall insulation, lath and plaster, painting, interior masonry, and air conditioning enclosures. The network shown in Figure 5 is a typical tier. Tier 8 has been chosen for illustrative purposes.

The activity input deck for subnet (TIER8) is shown in Figure 6. The reporting date is June 16, 1969, and the scheduled start date for the network was chosen as June 1, 1969.

The first report, shown in Figure 7, is report number 1 for subnet (TIER8). The heading contains the following information: run number, report date, sorting technique, network name, and which subnet. This sort is in the same order as input. Events which are interfaces have an "F" next to them, and events which are ends have an "E" next to them. The first column contains the predecessor event number. The second column contains the successor event number. The third column contains descriptions of the activities. Dummy activities are restraints in the network. The restraints are necessary when one activity is precedent to two or more activities as previously described. The fourth column contains the activities' most likely times. This information was input to the program. The fifth column contains the expected date. The expected date is program generated, and indicates the earliest date at which the activity may start. The sixth column contains the allowed date; this date is also program calculated, and indicates the latest date that the activity may start if the project is to finish on time. The seventh column contains the scheduled or actual date. This date is input to the program. In this example no scheduled dates were input. The eighth column contains the slack time in weeks and tenths of a week. These values are program generated. The ninth column contains the resource estimate. These values are input to the program. In this example the values are in one hundred dollar quantities. The tenth column contains the time remaining. This number is the number of weeks and tenths of weeks after the report date until the activity may be started. The last column contains the organization code. These numbers indicate which work unit is responsible for the activities' completion.

This report is useful if some logical numbering scheme was used to number the predecessor events. This sort is very useful to check input. If the input deck is out of order or a card is missing, it shows up as an error in this sort. This report is also useful as a reference to other sorts. Since this report is in input order, any data on subsequent reports can be cross-checked to determine if an error has been made during a sort.

The second report, shown in Figure 8, is report number 2 for subnet (TIER8). It contains the same information as report 1, but is sorted by successor event first and then by predecessor event.

The usefulness of this report parallels report number 1. A logical ordering of successor events can be checked with this sort.

The third report, shown in Figure 9, is report number 3 for subnet (TIER8). The same information as in reports 1 and 2 is given, but this report is sorted by slack. A double space between lines indicates a new value for slack. The eighth column contains the slack value. The first block of activities is the critical path.

This report is perhaps the most useful to the project supervisor. Since it is sorted by slack the most critical activities are at the top of this list, and the least critical activities are at the bottom. Using the resource column, decisions can be made to divert resources from the less critical activities to the more critical activities. In this way it may be possible to complete critical path activities in less time than their estimate. This would decrease total job duration. In the example shown in Figure 9 the activity of installing window units has 18.9 weeks of slack and requires $23,600.00 of resources. The resources are, to a large extent, labor. By utilizing a work force of less men, the activity would take longer to complete. In this way men could be freed to work on critical activities and lower overall project duration.

The fourth report, shown in Figure 10, is report number 4 for subnet (TIER8). This listing is sorted by expected date. This report contains the activities ordered by their expected start dates. This list indicates when activities may be started.

This report is also very useful in project control. It is sorted by expected date and indicates when resources will be needed. The report is useful to a project scheduler, who must order equipment, materials, and labor. This report is an efficient time schedule of when activities may take place. Report number 5, not shown, parallels report number 4, except that it is a schedule of when resources must be utilized.

The fifth report, shown in Figures 11 and 12, is report number 6 for subnet (TIER8), department 06. These reports are generated for all departments in tier 8. Department 06 was chosen as typical. This listing is all the activities that one department or organization is responsible for. This report also gives a monthly, quarterly, yearly, and total resource allocation breakdown. This report is extremely useful to inform an organization what resources are available. This report is also useful to monitor cost and budget data. Comparing predicted data with actual data, as the actual data becomes available, can detect budget overruns early. This data is available by department in a subnet as shown in Figure 11, and by subnet as shown in Figure 12.

The sixth report, shown in Figure 13, is report number 11 for subnet (TIER8), department 06. This report is sorted by department and by slack. It contains the advantages of both of those sorts as previously presented.

The ease of using this program has been shown. In calling these reports only one *REPORT card is needed. All the options may be specified on one card, and made in one run. The utility of the various reports has been demonstrated.

CHAPTER V

SUMMARY AND CONCLUSIONS

Systems analysis concepts and the techniques of network analysis have been reviewed briefly in order to establish a background against which the development of Program Evaluation and Review Technique (PERT) can be discussed. The evolution of PERT from its original development for the Polaris program through the current state-of-the-art, represented in NASA developed PERT TIME II, has also been reviewed.

Network analysis has been developed as a scheduling tool, and as such has been found to be an effective means of project control. In complex production and construction situations where continual project monitoring is required to assure the proper rate of progress, the development of PERT programs has filled a pressing need.

The capabilities of PERT TIME II have been discussed, with particular emphasis on the simplicity and utility of the computer program. The several user oriented features of this program, such as small amount of input, report variety, straight forward control mechanisms, and ease of updating make the program particularly attractive.

By making an application of PERT TIME II to the scheduling and construction control of a typical large building, it has been demonstrated that this NASA developed computer program can be implemented in the building construction industry. This application has demonstrated that such an aerospace systems analysis tool is readily adaptable to a segment of the construction industry. The usefulness of several features unique to PERT TIME II has been shown in this sample application.

It can be concluded that certain aspects of aerospace systems technology are readily adaptible and usefully applicable to the construction industry. In particular PERT TIME II, an advanced network analysis technique developed by NASA, holds promise for direct application to building construction projects.

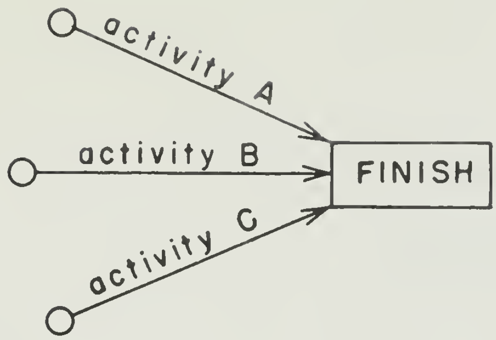


FIGURE 1

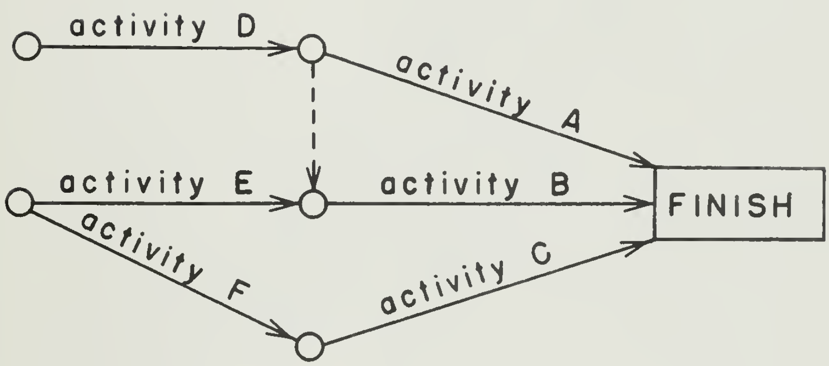


FIGURE 2

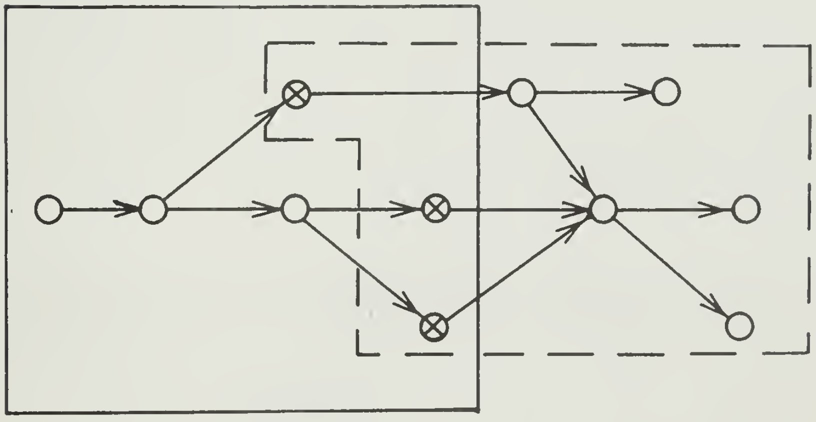


FIGURE 3

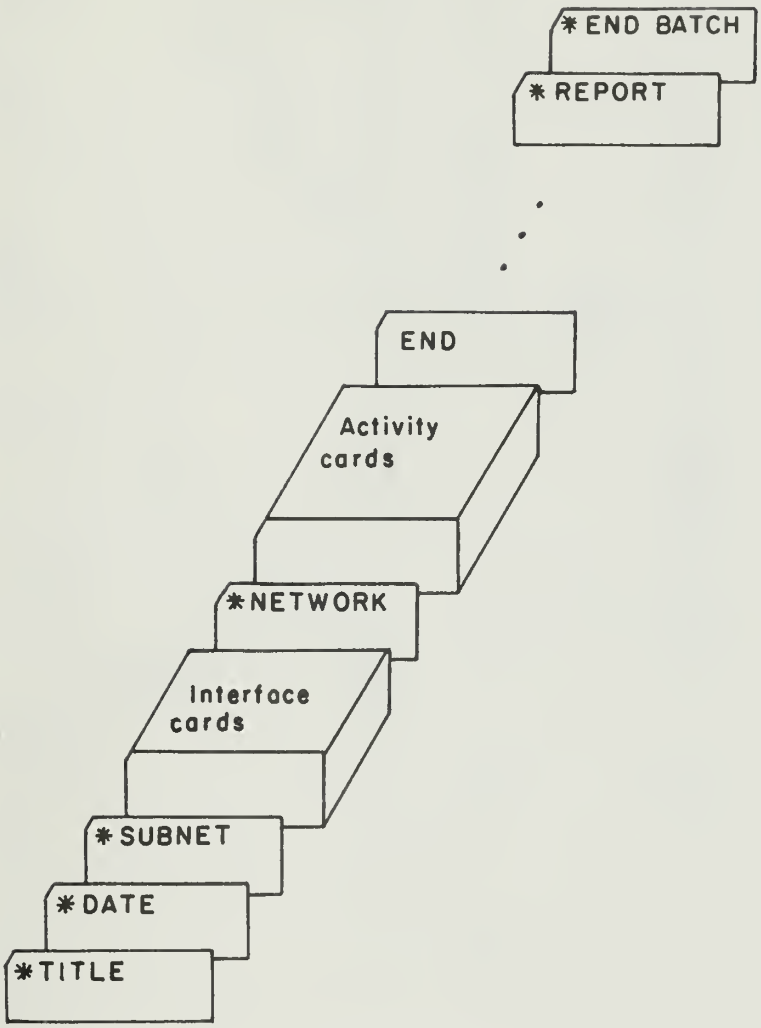
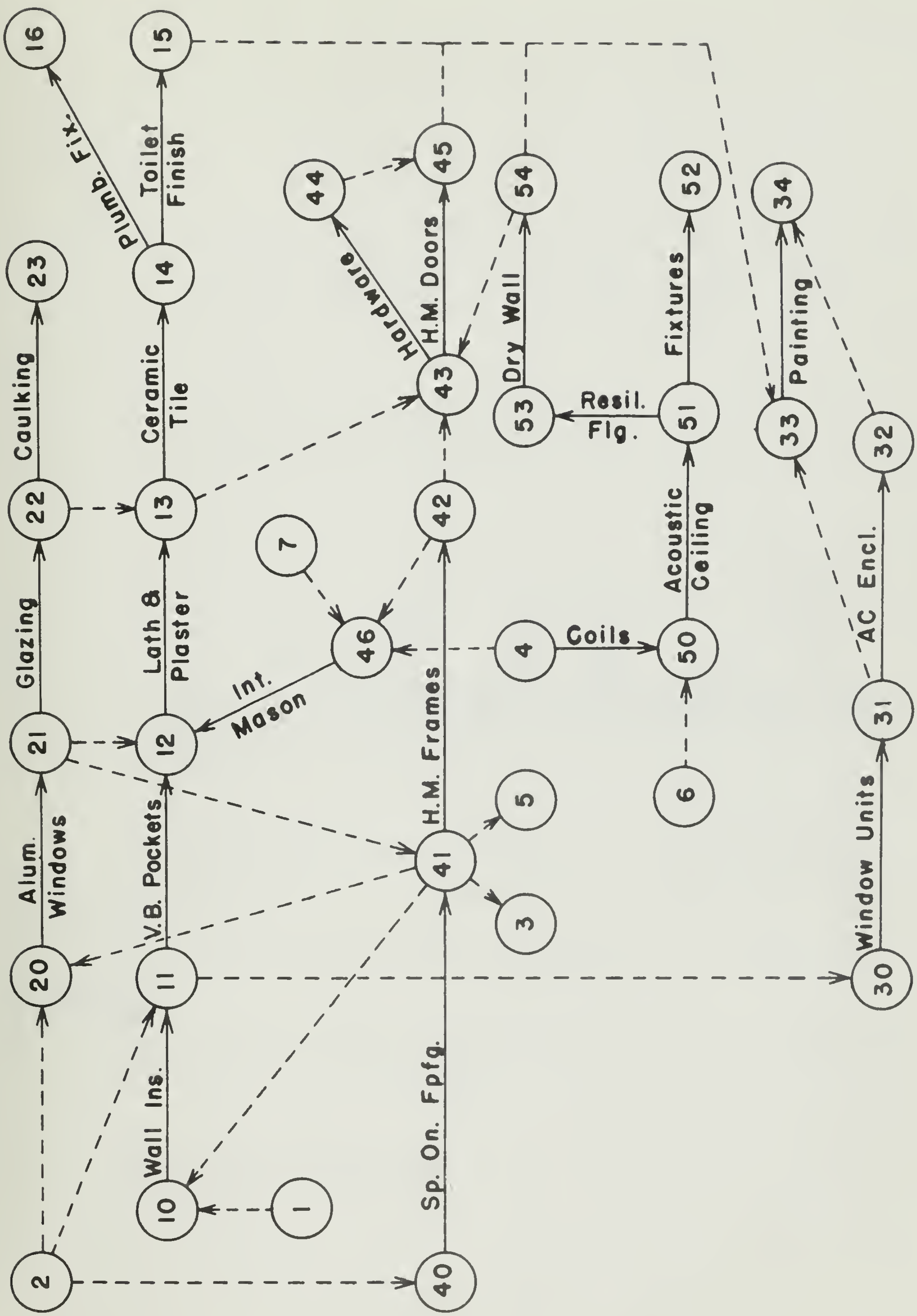

*END BATCH
*REPORT
END
Activity cards
*NETWORK
Interface cards
*SUBNET
*DATE
*TITLE

FIGURE 4



Alum. Windows
Glazing
Caulking
Wall Ins.
V.B. Pockets
Lath & Plaster
Ceramic Tile
Toilet Finish
Plumb. Fix.
Int. Mason
Sp. On. Fpfg.
H.M. Frames
H.M. Doors
Hardware
Dry Wall
Coils
Acoustic Ceiling
Resil. Flg.
Fixtures
Window Units
AC Encl.
Painting
1
2
3
4
5
6
7
10
11
12
13
14
15
16
20
21
22
23
30
31
32
33
34
40
41
42
43
44
45
46
50
51
52
53
54

FIGURE 5

```
*SUBNET (TIER8)
    F8G        1
    F9G        2
    F8D        3
    F9D        4
    F7W        5
    F10W       6
    C          7
*NETWORK
1        1     10  00    0     DUMMY
1        2     11  00    0     DUMMY
1        2     20  00    0     DUMMY
1        2     40  00    0     DUMMY
1        4     46  00    0     DUMMY
1        4     50  11   57     COILS                            07
1        6     50  00    0     DUMMY
1        7     46  00    0     DUMMY
1       10     11  17   62     WALL INSULATION                  01
1       11     12  17   72     V. B. POCKETS                    06
1       11     30  00    0     DUMMY
1       12     13  31  267     LATH + PLASTER                   02
1       13     14  14   72     CERAMIC TILE                     03
1       13     43  00    0     DUMMY
1       14     15  17   41     TOILET FINISH                    04
1       14     16  19   83     PLUMBING FIXTURES                04
1       15     33  00    0     DUMMY
1       20     21  26  284     ALUMINUM WINDOWS                 06
1       21     22  13  179     GLAZING                          06
1       21     12  00    0     DUMMY
1       22     23  11  190     CAULKING                         02
1       22     13  00    0     DUMMY
1       30     31  17  236     WINDOW UNITS                     07
1       31     32  17   71     A. C. ENCLOSURES                 07
1       31     33  00    0     DUMMY
1       32     34  00    0     DUMMY
1       33     34  14  211     PAINTING                         08
1       40     41  21  115     SPRAY ON FIREPROOFING
1       41     42  19   17     H. M. FRAMES                     06
1       41      3  00    0     DUMMY
1       41      5  00    0     DUMMY
1       41     10  00    0     DUMMY
1       41     20  00    0     DUMMY
1       42     43  00    0     DUMMY
1       42     46  00    0     DUMMY
1       43     44  19   70     HARDWARE                         06
1       43     45  11   16     H. M. DOORS                      06
1       44     45  00    0     DUMMY
1       45     33  00    0     DUMMY
1       46     12  21  617     INTERIOR MASONRY                 03
1       50     51  19  278     ACOUSTIC CEILING                 01
1       51     52  21  130     ELECTRICAL FIXTURES              05
1       51     53  19  150     RESILIENT FLOORING               01
1       53     54  14  565     DRYWALL                          02
1       54     33  00    0     DUMMY
1       54     43  00    0     DUMMY
END
```

FIGURE 6

NASA PERT TIME II
LEWIS RESEARCH CENTER

RUN 1

DATE OF THIS REPORT IS 6/16/69

BY PREDECESSOR AND SUCCESSOR EVENT NUMBER
NETWORK NEW ENGLAND MERCHANTS BANK

SUBNET TIER8

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	DATE SCHD/ACT.	SLACK	RE-SOURCE	TIME REM.	DEPT
0- 1F	0- 10	DUMMY	0.0	2/ 1/70	9/17/70		32.6	0	32.9	
0- 2F	0- 11	DUMMY	0.0	2/12/70	9/29/70		32.7	0	34.4	
0- 2F	0- 20	DUMMY	0.0	2/12/70	2/27/70		2.1	0	34.4	
0- 2F	0- 40	DUMMY	0.0	2/12/70	2/12/70		0.0	0	34.4	
0- 4F	0- 46	DUMMY	0.0	6/ 7/70	9/26/70		15.9	0	50.9	
0- 4F	0- 50	COILS	1.1	6/14/70	6/14/70		0.0	57	51.9	07
0- 6F	0- 50	DUMMY	0.0	4/ 1/70	6/14/70		10.6	0	41.3	
0- 7F	0- 46	DUMMY	0.0	9/26/70	9/26/70		0.0	0	66.7	
0- 10	0- 11	WALL INSULATION	1.7	3/10/70	9/29/70		28.9	62	38.1	01
0- 11	0- 12	V. B. POCKETS	1.7	3/22/70	10/11/70		28.5	72	39.9	06
0- 11	0- 30	DUMMY	0.0	3/10/70	11/ 9/70		34.8	0	38.1	
0- 12	0- 13	LATH & PLASTER	3.1	11/ 1/70	11/ 1/70		0.0	267	71.9	02
0- 13	0- 14	CERAMIC TILE	1.4	11/11/70	11/11/70		0.0	72	73.3	03
0- 13	0- 43	DUMMY	0.0	11/ 1/70	11/10/70		1.2	0	71.9	
0- 14	0- 15	TOILET FINISH	1.7	11/23/70	11/23/70		0.0	41	75.0	04
0- 14	0- 16E	PLUMBING FIXTURES	1.9	11/24/70	11/24/70		0.0	83	75.1	04
0- 15	0- 33	DUMMY	0.0	11/23/70	11/23/70		0.0	0	75.0	
0- 20	0- 21	ALUMINUM WINDOWS	2.6	3/17/70	3/17/70		0.0	284	35.1	06
0- 21	0- 22	GLAZING	1.3	3/26/70	3/26/70		0.0	179	40.4	06
0- 21	0- 12	DUMMY	0.0	3/17/70	10/11/70		29.7	0	39.1	
0- 22	0- 23E	CAULKING	1.1	4/ 3/70	4/ 3/70		0.0	190	41.6	02
0- 22	0- 13	DUMMY	0.0	3/26/70	11/ 1/70		31.5	0	40.4	
0- 30	0- 31	WINDOW UNITS	1.7	3/22/70	11/21/70		34.8	236	39.9	07
0- 31	0- 32	A. C. ENCLOSURES	1.7	4/ 3/70	12/ 3/70		34.8	71	41.6	07
0- 31	0- 33	DUMMY	0.0	3/22/70	11/23/70		35.1	0	39.9	
0- 32	0- 34E	DUMMY	0.0	4/ 3/70	12/ 3/70		34.8	0	41.6	
0- 33	0- 34E	PAINTING	1.4	12/ 3/70	12/ 3/70		0.0	211	76.4	08
0- 40	0- 41	SPRAY ON FIREPROOFING	2.1	2/27/70	2/27/70		0.0	115	36.6	
0- 41	0- 42	H. M. FRAMES	1.9	3/12/70	9/26/70		28.3	17	38.4	06
0- 41	0- 3F	DUMMY	0.0	2/27/70	5/30/70		13.2	0	36.6	
0- 41	0- 5F	DUMMY	0.0	2/27/70	5/22/70		12.1	0	36.6	
0- 41	0- 10	DUMMY	0.0	2/27/70	9/17/70		28.9	0	36.6	
0- 41	0- 20	DUMMY	0.0	2/27/70	2/27/70		0.0	0	36.6	
0- 42	0- 43	DUMMY	0.0	3/12/70	11/10/70		34.7	0	38.4	
0- 42	0- 46	DUMMY	0.0	3/12/70	9/26/70		28.3	0	38.4	
0- 43	0- 44	HARDWARE	1.9	11/15/70	11/23/70		1.2	70	73.9	06
0- 43	0- 45	H. M. DOORS	1.1	11/ 9/70	11/23/70		2.0	16	73.0	06
0- 44	0- 45	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 45	0- 33	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 46	0- 12	INTERIOR MASONRY	2.1	10/11/70	10/11/70		0.0	617	68.9	03
0- 50	0- 51	ACOUSTIC CEILING	1.9	6/28/70	6/28/70		0.0	278	53.9	01
0- 51	0- 52E	ELECTRICAL FIXTURES	2.1	7/12/70	7/12/70		0.0	130	55.9	05
0- 51	0- 53	RESILIENT FLOORING	1.9	7/11/70	10/31/70		16.0	150	55.7	01
0- 53	0- 54	DRYWALL	1.4	7/21/70	11/10/70		16.0	565	57.1	02
0- 54	0- 33	DUMMY	0.0	7/21/70	11/23/70		17.9	0	57.1	
0- 54	0- 43	DUMMY	0.0	7/21/70	11/10/70		16.0	0	57.1	

FIGURE 7

NASA PERT TIME II
LEWIS RESEARCH CENTER PAGE 1

RUN 1 DATE OF THIS REPORT IS 6/16/69

BY SUCCESSOR AND PREDECESSOR EVENT NUMBER
NETWORK NEW ENGLAND MERCHANTS BANK SUBNET TIER8

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	DATE SCHD/ACT.	SLACK	RE-SOURCE	TIME REM.	DEPT
0- 41	0- 3F	DUMMY	0.0	2/27/70	5/30/70		13.2	0	36.6	
0- 41	0- 5F	DUMMY	0.0	2/27/70	5/22/70		12.1	0	36.6	
0- 1F	0- 10	DUMMY	0.0	2/ 1/70	9/17/70		32.6	0	32.9	
0- 41	0- 10	DUMMY	0.0	2/27/70	9/17/70		28.9	0	36.6	
0- 2F	0- 11	DUMMY	0.0	2/12/70	9/29/70		32.7	0	34.4	
0- 10	0- 11	WALL INSULATION	1.7	3/10/70	9/29/70		28.9	62	38.1	01
0- 11	0- 12	V. P. POCKETS	1.7	3/22/70	10/11/70		28.9	72	39.9	06
0- 21	0- 12	DUMMY	0.0	3/17/70	10/11/70		29.7	0	39.1	
0- 46	0- 12	INTERIOR MASONRY	2.1	10/11/70	10/11/70		0.0	617	68.9	03
0- 12	0- 13	LATH & PLASTER	3.1	11/ 1/70	11/ 1/70		0.0	267	71.9	02
0- 22	0- 13	DUMMY	0.0	3/26/70	11/ 1/70		31.5	0	40.4	
0- 13	0- 14	CERAMIC TILE	1.4	11/11/70	11/11/70		0.0	72	73.3	03
0- 14	0- 15	TOILET FINISH	1.7	11/23/70	11/23/70		0.0	41	75.0	04
0- 14	0- 16E	PLUMBING FIXTURES	1.9	11/24/70	11/24/70		0.0	83	75.1	04
0- 2F	0- 20	DUMMY	0.0	2/12/70	2/27/70		2.1	0	34.4	
0- 41	0- 20	DUMMY	0.0	2/27/70	2/27/70		0.0	0	36.6	
0- 20	0- 21	ALUMINUM WINDOWS	2.6	3/17/70	3/17/70		0.0	284	39.1	06
0- 21	0- 22	GLAZING	1.3	3/26/70	3/26/70		0.0	179	40.4	06
0- 22	0- 23F	CAULKING	1.1	4/ 3/70	4/ 3/70		0.0	190	41.6	02
0- 11	0- 30	DUMMY	0.0	3/10/70	11/ 9/70		34.8	0	38.1	
0- 30	0- 31	WINDOW UNITS	1.7	3/22/70	11/21/70		34.8	236	39.9	07
0- 31	0- 32	A. C. ENCLOSURES	1.7	4/ 3/70	12/ 3/70		34.8	71	41.6	07
0- 15	0- 33	DUMMY	0.0	11/23/70	11/23/70		0.0	0	75.0	
0- 31	0- 33	DUMMY	0.0	3/22/70	11/23/70		35.1	0	39.9	
0- 45	0- 33	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 54	0- 33	DUMMY	0.0	7/21/70	11/23/70		17.9	0	57.1	
0- 32	0- 34E	DUMMY	0.0	4/ 3/70	12/ 3/70		34.8	0	41.6	
0- 33	0- 34E	PAINTING	1.4	12/ 3/70	12/ 3/70		0.0	211	76.4	08
0- 2F	0- 40	DUMMY	0.0	2/12/70	2/12/70		0.0	0	34.4	
0- 40	0- 41	SPRAY ON FIREPROOFING	2.1	2/27/70	2/27/70		0.0	115	36.6	
0- 41	0- 42	H. M. FRAMES	1.9	3/12/70	9/26/70		28.3	17	38.4	06
0- 13	0- 43	DUMMY	0.0	11/ 1/70	11/10/70		1.2	0	71.9	
0- 42	0- 43	DUMMY	0.0	3/12/70	11/10/70		34.7	0	38.4	
0- 54	0- 43	DUMMY	0.0	7/21/70	11/10/70		16.0	0	57.1	
0- 43	0- 44	HARDWARE	1.9	11/15/70	11/23/70		1.2	70	73.9	06
0- 43	0- 45	H. M. DOORS	1.1	11/ 9/70	11/23/70		2.0	16	73.0	06
0- 44	0- 45	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 4F	0- 46	DUMMY	0.0	6/ 7/70	9/26/70		15.9	0	50.9	
0- 7F	0- 46	DUMMY	0.0	9/26/70	9/26/70		0.0	0	66.7	
0- 42	0- 46	DUMMY	0.0	3/12/70	9/26/70		28.3	0	38.4	
0- 4F	0- 50	COILS	1.1	6/14/70	6/14/70		0.0	57	51.9	07
0- 6F	0- 50	DUMMY	0.0	4/ 1/70	6/14/70		10.6	0	41.3	
0- 50	0- 51	ACOUSTIC CEILING	1.9	6/28/70	6/28/70		0.0	278	53.9	01
0- 51	0- 52E	ELECTRICAL FIXTURES	2.1	7/12/70	7/12/70		0.0	130	55.9	05
0- 51	0- 53	RESILIENT FLOORING	1.9	7/11/70	10/31/70		16.0	150	55.7	01
0- 53	0- 54	DRYWALL	1.4	7/21/70	11/10/70		16.0	565	57.1	02

FIGURE 8

RUN 1 DATE OF THIS REPORT IS 6/16/69

BY PATHS OF CRITICALITY
NETWORK NEW ENGLAND MERCHANTS BANK

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	SUPNET DATE SCHD/ACT.	TIERA SLACK	RE-SOURCE	TIME REM.	DEPT
0- 2F	0- 40	DUMMY	0.0	2/12/70	2/12/70		0.0	0	34.4	
0- 41	0- 20	DUMMY	0.0	2/27/70	2/27/70		0.0	0	36.6	
0- 40	0- 41	SPRAY ON FIREPROOFING	2.1	2/27/70	2/27/70		0.0	115	36.6	
0- 20	0- 21	ALUMINUM WINDOWS	2.5	3/17/70	3/17/70		0.0	284	39.1	06
0- 21	0- 22	GLAZING	1.3	3/26/70	3/26/70		0.0	179	40.4	06
0- 22	0- 23F	CAULKING	1.1	4/ 3/70	4/ 3/70		0.0	190	41.6	02
0- 4F	0- 50	COILS	1.1	6/14/70	6/14/70		0.0	57	51.9	07
0- 50	0- 51	ACOUSTIC CEILING	1.5	6/28/70	6/28/70		0.0	278	53.9	01
0- 51	0- 52F	ELECTRICAL FIXTURES	2.1	7/12/70	7/12/70		0.0	130	55.9	05
0- 7F	0- 46	DUMMY	0.0	9/26/70	9/26/70		0.0	0	66.7	
0- 46	0- 12	INTERIOR MASONRY	2.1	10/11/70	10/11/70		0.0	617	68.9	03
0- 12	0- 13	LATH & PLASTER	3.1	11/ 1/70	11/ 1/70		0.0	267	71.9	02
0- 13	0- 14	CERAMIC TILE	1.4	11/11/70	11/11/70		0.0	72	73.3	03
0- 14	0- 15	TOILET FINISH	1.7	11/23/70	11/23/70		0.0	41	75.0	04
0- 15	0- 33	DUMMY	0.0	11/23/70	11/23/70		0.0	0	75.0	
0- 14	0- 16E	PLUMBING FIXTURES	1.9	11/24/70	11/24/70		0.0	83	75.1	04
0- 33	0- 34E	PAINTING	1.4	12/ 3/70	12/ 3/70		0.0	211	76.4	08
0- 13	0- 43	DUMMY	0.0	11/ 1/70	11/10/70		1.2	0	71.9	
0- 44	0- 45	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 45	0- 33	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 43	0- 44	HARDWARE	1.9	11/15/70	11/23/70		1.2	70	73.9	06
0- 43	0- 45	H. M. DOORS	1.1	11/ 5/70	11/23/70		2.6	16	73.0	06
0- 2F	0- 20	DUMMY	0.0	2/12/70	2/27/70		2.1	0	34.4	
0- 6F	0- 50	DUMMY	0.0	4/ 1/70	6/14/70		10.6	0	41.3	
0- 41	0- 5F	DUMMY	0.0	2/27/70	5/22/70		12.1	0	36.6	
0- 41	0- 3F	DUMMY	0.0	2/27/70	5/30/70		13.2	0	36.6	
0- 4F	0- 46	DUMMY	0.0	6/ 7/70	9/26/70		15.9	0	50.9	
0- 51	0- 53	RESILIENT FLOORING	1.9	7/11/70	10/31/70		16.0	190	55.7	01
0- 53	0- 54	DRYWALL	1.4	7/21/70	11/10/70		16.0	565	57.1	02
0- 54	0- 43	DUMMY	0.0	7/21/70	11/10/70		16.0	0	57.1	
0- 54	0- 33	DUMMY	0.0	7/21/70	11/23/70		17.9	0	57.1	
0- 42	0- 46	DUMMY	0.0	3/12/70	9/26/70		28.3	0	38.4	
0- 41	0- 42	H. M. FRAMES	1.9	3/12/70	9/26/70		28.3	17	38.4	06
0- 41	0- 10	DUMMY	0.0	2/27/70	9/17/70		28.9	0	36.6	
0- 10	0- 11	WALL INSULATION	1.7	3/10/70	9/29/70		28.9	62	38.1	01
0- 11	0- 12	V. B. POCKETS	1.7	3/22/70	10/11/70		28.9	72	39.9	06
0- 21	0- 12	DUMMY	0.0	3/17/70	10/11/70		29.7	0	39.1	

FIGURE 9

-44-

NASA PERT TIME II
LEWIS RESEARCH CENTER PAGE 2

RUN 1 DATE OF THIS REPORT IS 6/16/69

BY PATHS OF CRITICALITY
NETWORK NEW ENGLAND MERCHANTS BANK

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	SUPNET DATE SCHD/ACT.	TIERA SLACK	RE-SOURCE	TIME REM.	DEPT
0- 22	0- 13	DUMMY	0.0	3/26/70	11/ 1/70		31.0	0	40.4	
0- 1F	0- 10	DUMMY	0.0	2/ 1/70	5/17/70		32.6	0	32.9	
0- 2F	0- 11	DUMMY	0.0	2/12/70	5/29/70		32.7	0	34.4	
0- 42	0- 43	DUMMY	0.0	3/12/70	11/10/70		34.1	0	38.4	
0- 11	0- 30	DUMMY	0.0	3/10/70	11/ 5/70		34.6	0	36.1	
0- 30	0- 31	WINDOW UNITS	1.7	3/22/70	11/21/70		34.6	236	35.9	07
0- 32	0- 34F	DUMMY	0.0	4/ 3/70	12/ 3/70		34.6	0	41.6	
0- 31	0- 32	A. C. ENCLOSURES	1.7	4/ 3/70	12/ 3/70		34.6	71	41.6	07
0- 31	0- 33	DUMMY	0.0	3/22/70	11/23/70		35.1	0	35.9	

RUN 1

DATE OF THIS REPORT IS 6/16/69

BY EXPECTED DATE AND PREDECESSOR EVENT NUMBER
NETWORK NEW ENGLAND MERCHANTS BANK

SUBNET TIER8

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	DATE SCHD/ACT.	SLACK	RE-SOURCE	TIME REM.	DEPT
0- 1F	0- 10	DUMMY	0.0	2/ 1/70	5/17/70		32.6	0	32.9	
0- 2F	0- 11	DUMMY	0.0	2/12/70	5/29/70		32.7	0	34.4	
0- 2F	0- 20	DUMMY	0.0	2/12/70	2/27/70		2.1	0	34.4	
0- 2F	0- 40	DUMMY	0.0	2/12/70	2/12/70		0.0	0	34.4	
0- 40	0- 41	SPRAY ON FIREPROOFING	2.1	2/27/70	2/27/70		0.0	115	36.6	
0- 41	0- 3F	DUMMY	0.0	2/27/70	5/30/70		13.2	0	36.6	
0- 41	0- 5F	DUMMY	0.0	2/27/70	5/22/70		12.1	0	36.6	
0- 41	0- 10	DUMMY	0.0	2/27/70	5/17/70		28.5	0	36.6	
0- 41	0- 20	DUMMY	0.0	2/27/70	2/27/70		0.0	0	36.6	
0- 10	0- 11	WALL INSULATION	1.7	3/10/70	9/29/70		28.9	62	38.1	01
0- 11	0- 30	DUMMY	0.0	3/10/70	11/ 9/70		34.8	0	38.1	
0- 41	0- 42	H. M. FRAMES	1.5	3/12/70	9/26/70		28.3	17	38.4	06
0- 42	0- 43	DUMMY	0.0	3/12/70	11/10/70		34.7	0	38.4	
0- 42	0- 46	DUMMY	0.0	3/12/70	5/26/70		28.3	0	38.4	
0- 20	0- 21	ALUMINUM WINDOWS	2.6	3/17/70	3/17/70		0.0	284	39.1	06
0- 21	0- 12	DUMMY	0.0	3/17/70	10/11/70		29.7	0	39.1	
0- 11	0- 12	V. B. POCKETS	1.7	3/22/70	10/11/70		28.9	72	39.9	06
0- 30	0- 31	WINDOW UNITS	1.7	3/22/70	11/21/70		34.8	236	39.9	07
0- 31	0- 33	DUMMY	0.0	3/22/70	11/23/70		35.1	0	39.9	
0- 21	0- 22	GLAZING	1.3	3/26/70	3/26/70		0.0	175	40.4	06
0- 22	0- 13	DUMMY	0.0	3/26/70	11/ 1/70		31.5	0	40.4	
0- 6F	0- 50	DUMMY	0.0	4/ 1/70	6/14/70		10.6	0	41.3	
0- 22	0- 23E	CAULKING	1.1	4/ 3/70	4/ 3/70		0.0	190	41.6	02
0- 31	0- 32	A. C. ENCLOSURES	1.7	4/ 3/70	12/ 3/70		34.8	71	41.6	07
0- 32	0- 34E	DUMMY	0.0	4/ 3/70	12/ 3/70		34.8	0	41.6	
0- 4F	0- 46	DUMMY	0.0	6/ 7/70	9/26/70		15.5	0	50.9	
0- 4F	0- 50	COILS	1.1	6/14/70	6/14/70		0.0	57	51.9	07
0- 50	0- 51	ACOUSTIC CEILING	1.5	6/28/70	6/28/70		0.0	278	53.9	01
0- 51	0- 53	RESILIENT FLOORING	1.5	7/11/70	10/31/70		16.0	150	55.7	01
0- 51	0- 52E	ELECTRICAL FIXTURES	2.1	7/12/70	7/12/70		0.0	130	55.9	05
0- 53	0- 54	DRYWALL	1.4	7/21/70	11/10/70		16.0	565	57.1	02
0- 54	0- 33	DUMMY	0.0	7/21/70	11/23/70		17.5	0	57.1	
0- 54	0- 43	DUMMY	0.0	7/21/70	11/10/70		16.0	0	57.1	
0- 7F	0- 46	DUMMY	0.0	9/26/70	9/26/70		0.0	0	66.7	
0- 46	0- 12	INTERIOR MASONRY	2.1	10/11/70	10/11/70		0.0	617	68.9	03
0- 12	0- 13	LATH & PLASTER	3.1	11/ 1/70	11/ 1/70		0.0	267	71.9	02
0- 13	0- 43	DUMMY	0.0	11/ 1/70	11/10/70		1.2	0	71.9	
0- 43	0- 45	H. M. DOORS	1.1	11/ 5/70	11/23/70		2.0	16	73.0	06
0- 13	0- 14	CERAMIC TILE	1.4	11/11/70	11/11/70		0.0	72	73.3	03
0- 43	0- 44	HARDWARE	1.9	11/15/70	11/23/70		1.2	70	73.9	06
0- 44	0- 45	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 45	0- 33	DUMMY	0.0	11/15/70	11/23/70		1.2	0	73.9	
0- 14	0- 15	TOILET FINISH	1.7	11/23/70	11/23/70		0.0	41	75.0	04
0- 15	0- 33	DUMMY	0.0	11/23/70	11/23/70		0.0	0	75.0	
0- 14	0- 16E	PLUMBING FIXTURES	1.9	11/24/70	11/24/70		0.0	83	75.1	04
0- 33	0- 34E	PAINTING	1.4	12/ 3/70	12/ 3/70		0.0	211	76.4	08

FIGURE 10

NASA PERT TIME II
LEWIS RESEARCH CENTER

PAGE 7

RUN 1

DATE OF THIS REPORT IS 6/16/69

BY DEPARTMENT, EXPECTED DATE, AND PREDECESSOR EVENT
NETWORK NEW ENGLAND MERCHANTS BANK

SUBNET TIER8

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	DATE SCHD/ACT.	SLACK	RE-SOURCE	TIME REM.	DEPT
0- 41	0- 42	H. M. FRAMES	1.9	3/12/70	5/26/70		28.3	17	38.4	06
0- 20	0- 21	ALUMINUM WINDOWS	2.6	3/17/70	3/17/70		0.0	284	39.1	06
0- 11	0- 12	V. B. POCKETS	1.7	3/22/70	10/11/70		28.9	72	39.9	06
0- 21	0- 22	GLAZING	1.3	3/26/70	3/26/70		0.0	175	40.4	06
0- 43	0- 45	H. M. DOORS	1.1	11/ 9/70	11/23/70		2.0	16	73.0	06
0- 43	0- 44	HARDWARE	1.9	11/15/70	11/23/70		1.2	70	73.9	06

FIGURE II

NASA
LEWIS RESEARCH CENTER

RESOURCE REPORT

NEW ENGLAND MERCHANTS BANK
DISTRIBUTION BY ALLOWED DATE

TOTAL RESOURCES ALLOCATED FOR
WORK ACCOMPLISHED THROUGH 6/16/69 . 0

LAG TIME = 0

FY YEAR	JULY	AUG	SEPT	OCT	NOV	DEC	JAN	FEB	MARCH	APRIL	MAY	JUNE	1 ST QTR	2 ND QTR	3 RD QTR	4 TH QTR	TOTAL
1970	0	0	0	0	0	0	0	16	447	0	0	0	0	0	463	0	463
1971	0	0	17	72	86	0	0	0	0	0	0	0	17	158	0	0	175
																TOTAL	638

TOTAL RESOURCE INPUT FOR ABOVE REPORT WAS 638

NASA PERT TIME II
LEWIS RESEARCH CENTER

PAGE 7

RUN 1

DATE OF THIS REPORT IS 6/16/69

BY DEPARTMENT, EXPECTED DATE, AND PREDECESSOR EVENT
NETWORK NEW ENGLAND MERCHANTS BANK

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	SUBNET DATE SCHD/ACT.	TIER8 SLACK	RE-SOURCE	TIME REM.	DEPT
0- 41	0- 42	H. M. FRAMES	1.9	3/12/70	5/26/70		28.3	17	38.4	06
0- 20	0- 21	ALUMINUM WINDOWS	2.6	3/17/70	3/17/70		0.0	284	39.1	06
0- 11	0- 12	V. B. POCKETS	1.7	3/22/70	10/11/70		28.9	72	39.9	06
0- 21	0- 22	GLAZING	1.3	3/26/70	3/26/70		0.0	175	40.4	06
0- 43	0- 45	H. M. DOORS	1.1	11/ 9/70	11/23/70		2.0	16	73.0	06
0- 43	0- 44	HARDWARE	1.9	11/15/70	11/23/70		1.2	70	73.9	06

FIGURE II

NASA
LEWIS RESEARCH CENTER

RESOURCE REPORT

TOTAL RESOURCES ALLOCATED FOR
WORK ACCOMPLISHED THROUGH 6/16/69 = 0

NEW ENGLAND MERCHANTS BANK
DISTRIBUTION BY ALLOWED DATE

LAG TIME = 0

FY YEAR	JULY	AUG	SEPT	OCT	NOV	DEC	JAN	FEB	MARCH	APRIL	MAY	JUNE	1 ST QTR	2 ND QTR	3 RD QTR	4 TH QTR	TOTAL
1970	0	0	0	0	0	0	0	16	447	0	0	0	0	0	463	0	463
1971	0	0	17	72	86	0	0	0	0	0	0	0	17	158	0	0	175
																TOTAL	638

TOTAL RESOURCE INPUT FOR ABOVE REPORT WAS 638

NASA
LEWIS RESEARCH CENTER

RESCURCE REPCRT

*** SUMMARY ***

NEW ENGLAND MERCHANTS BANK
CISTRIBUTICN BY ALLCWEC CATE

TOTAL RESCURCES ALLCCATED FCR
WORK ACCCMPLISHED THROUGH 6/16/69 = C

LAG TIME = 0

FY YEAR	JULY	AUG	SEPT	OCT	NOV	CEC	JAN	FEB	MARCH	APRIL	MAY	JUNE	1 ST QTR	2 CND QTR	3 RC QTR	4 TH QTR	TCTAL
1970	0	0	C	C	C	0	0	131	556	81	0	354	0	C	687	435	1122
1971	111	C	211	961	1288	90	0	0	0	C	0	0	322	2339	C	C	2661
																TCTAL	3783

TOTAL RESCURCE INPUT FCR ABCVE REPCRT WAS 3783

FIGURE 12

NASA PERT TIME II
LEWIS RESEARCH CENTER

PAGE 7

RUN 1

DATE OF THIS REPORT IS 6/16/69

BY DEPARTMENT AND SLACK
NETWORK NEW ENGLAND MERCHANTS BANK

EVENT PRE.	SUC.	ACTIVITY DESCRIPTION	ACTIV. TIME	DATE EXPECTED	DATE ALLOWED	SUBNET DATE SCHD/ACT.	TIER8 SLACK	RE-SOURCE	TIME REM.	DEPT
0- 20	0- 21	ALUMINUM WINDOWS	2.6	3/17/70	3/17/70		0.0	284	39.1	06
0- 21	0- 22	GLAZING	1.3	3/26/70	3/26/70		0.0	179	40.4	06
0- 43	0- 44	HARDWARE	1.9	11/15/70	11/23/70		1.2	70	73.9	06
0- 43	0- 45	H. M. DOORS	1.1	11/ 9/70	11/23/70		2.0	16	73.0	06
0- 41	0- 42	H. M. FRAMES	1.9	3/12/70	9/26/70		28.3	17	38.4	06
0- 11	0- 12	V. B. POCKETS	1.7	3/22/70	10/11/70		28.9	72	39.9	06

FIGURE 13

BIBLIOGRAPHY

1 Tyler, V. M., and C. F. Asiala, "Aerospace Role in Planning Cities for the Future", Paper 66-878, November, 1966, AIAA.

2 Avdeev, Y. A., and A. P. Nikolaeva, "Project Control Using the Critical Path Technique", Computer Elements and Systems, Nos. 10-14, pages 92-114, 1964-1966.

3 "Systems Building: What It Really Means", Architectural Record, pages 147-154, January, 1969.

4 Bremer, Hans, "Topological Ordering Using the Pushdown Technique", Proc. NASA/Industry PERT computer Conference, 1964.

5 "NASA PERT and Companion Cost System Handbook", United States Government Printing Office, 1962.

6 Hirsh, D. J., F. Funicelli, and J. D. Walker, "NASA PERT TIME II User's Manual", Lewis Research Center, 1966.

7 Seelig, W. D., and I. M. Rubin, "The Effects of PERT on Research and Development Organizations", M. I. T. Sloan School of Management, Paper 230-66, 1966.

8 Hilton, M. W., "The Use of PERT in Industry", Master's Thesis, M. I. T., Sloan School of Management, 1966.

9 Antill, J. M., and R. W. Woodhead, "Critical Path Methods in Construction Practice", John Wiley & Sons, 1965.

10 Chestnut, H., "Systems Engineering Methods", John Wiley & Sons, 1967.

APPENDIX A

GLOSSARY

ACTIVITY: A time consuming element in the network. It is represented as an arrow, and defined by a starting (predecessor) event and an ending (successor) event.

ALLOWABLE TIME: The latest possible time that an event can be allowed to occur.

CRITICAL PATH OR PACING PATH: The sequence of activities which make up the most stringent time constraint. The path with the smallest amount of positive slack or largest amount of negative slack.

CONTROL NETWORK: The network formed by all control network activities and events.

CONTROL NETWORK ACTIVITY: The single activity made up by condensing all activities between two control network events.

CONTROL NETWORK EVENT: All start, end, and interface events relative to a particular collection of subnets.

EVENT: An identifiable instant of time which is meaningful specified accomplishment.

EXPECTED TIME: The earliest possible time that an event can be expected to occur.

FRAGNET: An identifiable portion of the total network as seen by the analyst.

INTERFACE EVENT: An event common to two or more subnets or fragnets.

NETWORK: The collection of all events and activities needed to accomplish the network objective.

PREDECESSOR: An event immediately preceding a given activity.

SLACK: The difference of time between predecessor event and successor event and activity time. Positive slack indicates an excess amount of time to complete an activity and negative slack indicates time which is not available to complete an activity.

SUBNET: An identifiable portion of the total network as seen by the computer.

SUCCESSOR: An event immediately following a given activity.

APPENDIX B

LIST OF FIGURES

APPENDIX C

USER'S MANUAL

The input to the PERT TIME II program is problem oriented. That is most control words are written in free format. The control words are self-explanatory. The following user's manual describes the words and their options used in the program. How typical decks of cards are set up is also presented. Report generation is developed in detail. This manual is sufficient to run PERT TIME II programs.

For ease of coordination it is not necessary for each subnet to have a unique numbering system. Furthermore, it is not necessary for interface events to be numbered the same in different subnets. In the network as a whole, the interface has an alphanumeric name, but in the individual subnets it may have any number.

Monitor control cards depend on the computer installation. These cards initiate a run, then program control cards must follow prior to the network. The format for the monitor cards is available at the computer installation. The format for the program control cards, and their order, is given below.

1. <u>Input Features</u>

a. Events may be randomly numbered without any sequential order along a path. The same number may be used in any number of fraqnets.

b. End events should have schedule dates. If no date is given, the expected date is used.

c. A complete activity or a code 4 schedule start should be used as the starting activity for a network. If this procedure is not followed, an error message is generated and the internal base date, 12/31/56, is used for calculations.

2. Program Control Cards

a. There are two kinds of program control cards:

1. Standard program control cards which must be used.

2. Optional program cards which are necessary for the various program options.

b. The general rules for program control cards:

1. Free format except that the asterisk must be punched in card column 1.

2. Parentheses, slashes, and commas are necessary where indicated.

3. Brackets [] indicate required information and braces {} indicate optional information.

4. Up to 13 options may be punched on a single control card. Options are separated by commas.

The program control cards are:

*TITLE [(Project Name)], is a standard card which provides an input for the project name. This card must follow the monitor control cards.

*DATE [(month/day/year)], is a standard card which inputs the report date. Date must be numeric. This card must follow the monitor control cards.

*SUBNET [(Fragnet Name)], is a standard card which inputs a fragnet name and instructs the program that interface cards will follow. The name is limited to six alphanumeric characters. The fragnets interface cards must follow immediately.

*NETWORK, is a standard card which is fixed in format. It must be punched in card columns 1 through 8. This card instructs the program that activity cards will follow. The activity cards must follow immediately.

END, is a standard card which is fixed in format. It must be punched in card columns 1 through 3. It instructs the program that the end of the activities has been reached.

*END BATCH, is a standard card which indicates the end of a job. It is the last card in the deck.

*REPORT [(Fragnet Name)] {1,2,...14, punch}, is an optional card which specifies the reports desired for the fragnet indicated. These cards should be ordered in the same way as the fragnets and follow the last END card. This is not a fatal error, but an out of sequence message will be generated on the printout if the cards are not ordered. The PUNCH option will instruct the program to punch out a fragnet activity deck. Report 1 is automatically called for with the PUNCH option. Reports can be generated after each end run. In that case *REPORT cards also follow the *END RUN card and specify the desired reports for that end run for the given fragnets. If the same reports are desired on all end runs, only one set of *REPORT cards are needed immediately following the first *END RUN card.

*CONTROL {1,2,...14}, is an optional card which specifies reports to be generated on the control subnet.

*CUTOFF ({Critical Path = x,} {Expected Date = x,} {Allow Date = x}), is an optional card which specifies the number of critical paths and the required number of weeks of expected and/or allowed dates past the report date.

*RESOURCE ({EXP DATE or ALLOW DATE}, {LAG = x}), is an optional card which gives a resource report. The report can be obtained with an expected date or allowed date and a lag time. If no options are stated, the program uses the expected date with a zero lag time.

*COMPLETE ({PRINTOUT,} {HISTORY}), is an optional card which allows the user to maintain completed activities on the printout (PRINTOUT) and/or history cards are punched out (HISTORY).

*SUMMARY [(Fragnet Name)], is an optional card which identifies a fragnet as a summary fragnet. This card is used in place of the *SUBNET card and has the same characteristics.

*UPDATE, is an optional card which instructs the program to update the master file.

*DELETE [(Fragnet Name)], is an optional card which instructs the program to delete an entire fragnet from the master file. It must be in the same sequential order in the input as the fragnet is on the master file.

*INSERT [(Fragnet Name)] {SUMMARY}, is an optional card which instructs the program to add an entire fragnet to the existing master file. This card is used in place of the *SUBNET or *SUMMARY card. If it is a summary fragnet, it is identified {SUMMARY}.

*END POINT (XXXXX, XXXXX,...), is an optional card which asks for an end run. All end run interface names must be in parenthesis. Multiple cards may be used.

*END RUN [(Interface Name, month/day/year)], is an optional card which specifies the interface name and date for every end run desired. Each end run must have a separate card. It follows the *REPORT or *EXTRACT cards.

*EXTRACT [(Fragnet Name)] {1,2,...14}, is an optional card which instructs the program to extract the given fragnet from network and process it as an independent network. The reports desired are specified. This card follows the *REPORT cards. If there are no *REPORT cards, then the *EXTRACT card follows the END card for the last fragnet.

Interface and activity cards are fixed in format.

Columns	Interface Cards
4-8	Alphanumeric interface name punched anywhere in the given field.
9-15	Event numbers need not fill the entire field but must be right justified.
16-51	Interface event description may be punched anywhere in the given field.

Columns	Activity Cards
1	Code identifying the status: 1 establishes a new activity 2 re-estimate or change of activity data 3 activity has been completed, date must be supplied in columns 26-31 4 schedule start date, supplied in columns 26-31 5 delete activity
2,3	Master schedule flags. Punches indicate on which master schedule activity will be printed. May be left blank.
4-10	Predecessor event number, right justified. Unused portion of the field may be blank or zero.

11-17	Successor event number, right justified. Unused portion of the field may be blank or zero.
18-21	Time estimate: 18-20 for number of whole weeks, 21 for tenths of a week.
22-25	Resource data in the form of manhours, cost, and so forth. Right justified, may be left blank.
26-31	Actual or schedule date given by month, 26-27, day, 28-29, and year 30-31, and each right justified. May be blank unless 3 or 4 punched in column 1.
32-73	Activity description, may be left blank.
77-80	Organization code, may be left blank. Used for organization sorts.

Activity cards need only be sorted by predecessor event number for input.

The first cards in the deck are the monitor control cards. Next the program control cards, followed by the fragnet card groupings, and finally the report cards. In each fragnet grouping the interface cards come first, followed by the fragnet control cards, and then the activity cards. A typical deck might be set up as follows.

Monitor control cards

*TITLE

```
*DATE
*RESOURCE
*COMPLETE
*CONTROL
*SUBNET (A)
 interface cards for Subnet A
*NETWORK
 activity cards for Subnet A
END
*SUBNET (B)
 interface cards for Subnet B
*NETWORK
 activity cards for Subnet B
END
*REPORT (A)
*REPORT (B)
*END BATCH
```

3. Report Generation

Reports fall into two catagories: fragnet reports, and control network reports. Fragnet reports make up the largest portion of the output.

The fragnet reports desired are grouped together by fragnet. Before each fragnet group the title and statistics for that fragnet are printed. These statistics include:

a. Project network title

b. Number of activities in the fragnet

c. Number of events in the fragnet

d. Number of starts in the fragnet

e. Number of ends in the fragnet

f. Number of unscheduled ends in the fragnet

g. If the report is by extraction, it is so indicated.

An S, F, or E printed next to an event indicates a start event, interface event, or an end event, respectively.

The reports and their numbers are:

Number	Description of Report
1	By predecessor event number (By predecessor event number and successor event number if input that way).
2	By successor event number and predecessor event number.
3	By paths of criticality (This report is sorted by slack, expected date, and predecessor event number).
4	By expected date and predecessor event number.
5	By allowed date and predecessor event numbers.
6	By organization code (SSSS), expected date, and

predecessor event number.

7 By organization code (SSSN), expected date, and predecessor event number.

8 By organization code (SSNN), expected date, and predecessor event number.

9 By organization code (SNNN), expected date, and predecessor event number.

10 By organization code (SSSS), and successor event number.

11 By organization code (SSSS), and paths of criticality.

12 By master schedule (SS), expected date, and predecessor event number.

13 By master schedule (NS), expected date, and predecessor event number.

14 By master schedule (SN), expected date, and predecessor event number.

For report 6-14 the S's indicated sorted, the N's indicate not sorted for the columns designated. Reports 6-11 sort by organization code, columns 77 to 80 on the activity cards, and reports 12-14 sort by master schedule, columns 2 and 3 on the activity cards. For each change in characters being sorted the program starts a new page. Reports 6-9 have the resource report option. Reports 12-14 are generated only for activities with punches, other than zero, in columns 2 and 3.

The control network generation is the same as the fragnet reports except that no activity description is given. In place of the activity description the name of the fragnet from which the activity originated is given. The control network is internally generated.

The program outputs only the last completed activity on each path. By the use of the *COMPLETE control card completed activities may be output for recording purposes.

The resource report option is available with reports 6-9, and allows the user to correlate schedule and resources. This report is available on a fragnet basis only. The *RESOURCE control card instructs the program to distribute the resource, given on the activity card, linearly with time over the duration of the activity. The report totals the portion of resources expended through the reporting date. The total used is given in the heading, and resources to be expended appear in the body of the report.

The network need only be input once. Then only update cards need to be input. To update interface cards, a new card is typed in the same format as the old interface cards. The program reads the new card and compares it with the old interface cards. If the interface names do not match, then the information on the card is added to the file as a new interface. If the interface names match, then a check of the event numbers is made. If the event numbers match, the existing interface card is deleted.

To update the activity cards, use the same format as in the original activity cards. Note: code 1 is used to change information on an already existing activity. The program reads a 1 in card column 1 and then compares the predecessor and successor event numbers to the old file. If a match is found, the old activity is deleted, and the new one is used.

In order to insert a fragnet after an existing fragnet and that fragnet requires no updating, the following deck setup is used. Simply place in the update deck the *SUBNET or *SUMMARY card followed by an END card and then the *INSERT card and the fragnet to be inserted.

Originally the activity cards have to be input by predecessor event number and also successor event number if report 1 will be called and predecessor-successor event number order is desired. For subsequent file maintenance runs the update activity cards can be in any order.

Interface cards can be in any order originally and on update runs since the program will sort these events.

The deck setup for file maintenance is analogous to that of a regular card deck but there are some peculiarities.

1. In case only the interface cards are being updated, only the *SUBNET and END cards are needed in the fragnet card grouping.

2. If only the activities are being updated, all the control cards must be present, the *SUBNET, *NETWORK, and END cards.

3. To redesignate a fragnet as a summary fragnet without updating any interface or activity cards, the only control cards necessary are the *SUMMARY card followed by an END card.

4. No control cards are necessary for fragnets not being updated. If reports are desired for these fragnets, only their respective *REPORT cards are necessary.

5. To request reports without any updating in the network, the only cards necessary are the *TITLE, *DATE, *REPORT, and *END BATCH cards.

The program has a complete set of error diagnostics which are self-explanatory. Generally, the errors are limit type errors. The program continually checks for limit violations such as too many activities in a fragnet.